UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/767,681	01/29/2004	Gang Luo	11338 (NCR.0120US)	7097

26890 7590 02/08/2008

JAMES M. STOVER
TERADATA CORPORATION
2835 MIAMI VILLAGE DRIVE
MIAMISBURG, OH 45342

EXAMINER	
TRUONG, CAM Y T	

ART UNIT	PAPER NUMBER
2162	

MAIL DATE	DELIVERY MODE
02/08/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/767,681	LUO ET AL.	
	Examiner	Art Unit	
	Cam Y T. Truong	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 January 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-4,9-14,18 and 20-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-4,9-14,18 and 20-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Applicant has canceled claim 5 in the amendment filed on 1/23/2008.

Response to Arguments

2. Applicant's arguments, filed 1/23/2008, have been fully considered and are persuasive. The filed Final Office action on 11/23/2007 has been withdrawn.

Applicant's arguments with respect to claims 1-4, 9-15, 18 and 20-26 have been considered but are moot in view of the new ground(s) of rejection.

Applicant argued that Ganesh43 does not teach "partitioning received modification operations by submitting the modifications operations to different sessions based on which base relation the modification operations operate on".

Examiner respectfully disagrees. Ganesh43 teaches For the purposes of illustration, consider the following sequence of transactions which executes the indicated database statements (in SQL-based pseudocode) against a database table "Emp_Table" having the structure Emp_Table (emp_name, emp_value): (15) At commit time 5--Transaction T1 commits having executed the following statement: (16) INSERT INTO Emp_Table VALUES (`Smith`, X); (17) At commit time 10--Transaction T2 commits having executed the following statement: 18) INSERT INTO Emp_Table VALUES (`Jones`, Y); (19) At commit time 15--Transaction T3 commits having executed the following statements: (20) UPDATE Emp_Table SET Emp_value=X+1 WHERE emp_name=`Smith`; (21) UPDATE Emp_Table SET Emp_value=Y+1 WHERE emp_name=`Jones (col. 4,

lines 60-67; col. 5, lines 20). .The above information shows that statements are divided for execution for each commit time (as each session).

Applicant argued that there is no teaching that the modification operations that operate on a set of one or more tuples of a first base relation are grouped by a controller".

In response to applicant's argument, Ganesh's 943 teaches For the purposes of illustration, consider the following sequence of transactions which executes the indicated database statements (in SQL-based pseudocode) against a database table "Emp_Table" having the structure Emp_Table (emp_name, emp_value): (15) At commit time 5--Transaction T1 commits having executed the following statement: (16) INSERT INTO Emp_Table VALUES (`Smith`, X); (17) At commit time 10--Transaction T2 commits having executed the following statement: 18) INSERT INTO Emp_Table VALUES (`Jones`, Y); (19) At commit time 15--Transaction T3 commits having executed the following statements: (20) UPDATE Emp_Table SET Emp_value=X+1 WHERE emp_name=`Smith`; (21) UPDATE Emp_Table SET Emp_value=Y+1 WHERE emp_name=`Jones (col. 4, lines 60-67; col. 5, lines 20). .The above information shows that statements are divided by a controller for execution for each commit time.

As discussed above the combination of cited references teach the claimed invention.

Claim Objections

3. .Claims 20-22 are objected to because of the following informalities:

The phrase "adapted to" in claims 20-22, suggests or makes optional but does not require steps to be performed or does not limit a claim to a particular structure does not limit the scope of a claim or claim limitation. The following are examples of language that may raise a question

as to the limiting effect of the language in a claim:

(A) statements of intended use or field of use,

(B) "adapted to" or "adapted for" clauses,

(C) "wherein" clauses, or

(D) "whereby" clauses.

This list of examples is not intended to be exhaustive. See also MPEP § 2111.04.

Appropriate correction is required.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-4, 12-15, 22, 20 are rejected under 35 U.S.C. 103(a) being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view Ganesh et al (or hereinafter "Ganesh43") (US 6714943).

As to claim 1, Ganesh teaches the claimed limitations:

"a method for use with a database system that stores a join view associated with plural base relations" as (col. 3, lines 15-20), comprising:

"receiving modification operations that modify at least two of the base relations of the join view, wherein the at least two base relations comprise a first base relation and a second base relation" as (col. 3, lines 38-46);

"schedule transactions to avoid execution of modification operations of more than one of the at least two base relations at one time in the database system" as (col. 4, lines 15-45).

Ganesh does not explicitly teach the claimed limitations "performing partitioning of the received modifications operations by submitting at least some of the modification operations operating on the first base relation to a first session, and submitting at least another of the modification operations that operate on the second based relation to a second session; "grouping the at least some of the modification operations in the first session operating on the first base relation into a first transaction; wherein the at least another modification operation in the second session is part of a second transaction".

Ganesh43 teaches transaction T5 commits at time 25 having executed the following statements: UPDATE Emp_Table SET Emp_value=Emp_value+1 WHERE emp_name=`Smith`; DELETE FROM Emp_Table WHERE emp_name=`Miller'.

Statements are represented as modification operations that are grouped in the transaction T5 (col. 5, lines, 1-20, fig. 2). Transactions 1-5 are scheduled (col. 7, lines 22-40).

Ganesh's 43 teaches for the purposes of illustration, consider the following sequence of transactions which executes the indicated database statements (in SQL-based pseudocode) against a database table "Emp_Table" having the structure Emp_Table (emp_name, emp_value): (15) At commit time 5--Transaction T1 commits having executed the following statement: INSERT INTO Emp_Table VALUES (`Smith`, X); At commit time 10--Transaction T2 commits having executed the following statement: INSERT INTO Emp_Table VALUES (`Jones`, Y); At commit time 15--Transaction T3 commits having executed the following statements: (20) UPDATE Emp_Table SET Emp_value=X+1 WHERE emp_name=`Smith`; UPDATE Emp_Table SET Emp_value=Y+1 WHERE emp_name=`Jones (col. 4, lines 60-67; col. 5, lines 20). The above information shows that statements are divided for execution for each commit time as each session.

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ganesh43's teaching of assigning each transaction including statements to each commit time and transaction T5 commits at time 5 having executed the following statements to Ganesh's system in order to prevent conflict when multiple transactions trying to modify database records at the same time, minimize network traffic, achieve maximum performance (Ganesh49, col. 9, line 40-41) and further to allow users to track transactions in sessions in a sequence time.

As to claim 2, Ganesh teaches the claimed limitations:

"determining that the first transaction conflicts with the second modification operation based on the first and second transaction based on the first and second transactions modifying more than one base relation of the join view" as (fig. 6, col. 3, lines 38-67; col. 4, lines 1-35); and

"selecting one of first and second transaction for execution in the database system" as (col. 4, lines 15-45).

As to claim 3, Ganesh teaches the claimed limitations:

"wherein selecting one of the first and second transactions comprises selecting the first transaction" as (col. 4, lines 25-35).

Ganesh does not explicitly teach the claimed limitation "storing the second transaction in a queue".

Ganesh43 teaches storing transactions in a table as a queue (fig. 8). Since transaction TXA has a dependent SCN of "0", this transaction is not dependent upon any other transactions, and can be ordered before, after or parallel to any other transaction, subject to the ordering/dependency constraints of these other transactions. Transactions TXB, TXC, and TXE all have dep_SCN values of 5; therefore, these transactions must be scheduled to begin after all other transactions having SCN values of 5 or less have completed and committed (col. 16, lines 66-67; col. 17, lines 1-30).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ganesh43's teaching of storing transactions in a table, transactions must be scheduled to begin after all other transactions having SCN values of 5 or less have completed and committed and to Ganesh's system in order to order to scheduling the transactions for preventing conflict when multiple transactions trying to modify database records at the same time and further minimize network traffic and achieve maximum performance (col. 9, line 40-41).

As to claim 4, Ganesh does not explicitly teach the claimed limitation "waiting for the first transaction to complete execution before scheduling the second transaction for execution".

Ganesh43 teaches Since transaction TXA has a dependent SCN of "0", this transaction is not dependent upon any other transactions, and can be ordered before, after or parallel to any other transaction, subject to the ordering/dependency constraints of these other transactions. Transactions TXB, TXC, and TXE all have dep_SCN values of 5; therefore, these transactions must be scheduled to begin after all other transactions having SCN values of 5 or less have completed and committed (col. 16, lines 66-67; col. 17, lines 1-30).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ganesh43's teaching to Ganesh's system in order to order to scheduling the transactions for preventing conflict when multiple transactions

trying to modify database records at the same time and further minimize network traffic and achieve maximum performance (col. 9, line 40-41).

As to claim 12, Ganesh teaches the claimed limitations:

"receiving modification operations that modify at least two of the base relations of the join view, wherein the at least two base relations comprise a first base relation and a second base relation" as (col. 3, lines 38-46);

"schedule the transactions to avoid execution of transactions of more than one of the at least two base relations of the join view (col. 4, lines 15-45).

Ganesh does not explicitly teach the claimed limitation "perform partitioning of the received modification operations by submitting at least some of the modification operations operating on the first base relation to a first session, and submitting at least another of the modification operations that operate on a second base relation to a second session, group the at least some of the modification operations in the first session operating on the first base relation into a first transaction, wherein the at least another modification operation in the second session is part of a second transaction".

Ganesh's 43 teaches for the purposes of illustration, consider the following sequence of transactions which executes the indicated database statements (in SQL-based pseudocode) against a database table "Emp_Table" having the structure Emp_Table (emp_name, emp_value): (15) At commit time 5--Transaction T1 commits having executed the following statement: INSERT INTO Emp_Table VALUES (`Smith`,

X); At commit time 10--Transaction T2 commits having executed the following statement: INSERT INTO Emp_Table VALUES (`Jones`, Y); At commit time 15--Transaction T3 commits having executed the following statements: (20) UPDATE Emp_Table SET Emp_value=X+1 WHERE emp_name=`Smith`; UPDATE Emp_Table SET Emp_value=Y+1 WHERE emp_name=`Jones (col. 4, lines 60-67; col. 5, lines 20). The above information shows that statements are divided for execution for each commit time as each session.

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ganesh43's teaching of assigning each transaction including statements to each commit time and transaction T5 commits at time 25 having executed the following statements to Ganesh's system in order to prevent conflict when multiple transactions trying to modify database records at the same time, minimize network traffic, achieve maximum performance (Ganesh49, col. 9, line 40-41) and further to allow users to track transactions in sessions in a sequence time.

Claim 13 is rejected under the same reason as discussed in claim 2.

Claim 14 is rejected under the same reason as discussed in claim 3.

Claim 15 is rejected under the same reason as discussed in claim 4.

As to claim 22, Ganesh teaches the claimed limitations:

"a controller having one or more processor" as (col. 7, lines 5-20),

"to receive modification operations to modify plural base relations of a join view, the modification operations comprising modification operations to modify a first base relation of the join view, and modification operations to modify a second base relation of the join view" as col. 3, lines 38-46);

"re-order received modification operations to avoid concurrent execution of modification operations of more than one of the plural base relations of the join view" as (fig. 6, col. 3, lines 38-67; col. 4, lines 1-15);

"wherein certain of the modification operations on the first base relation comprise medication of set of one or more types of the first base relation" as (col. 4, lines 15-65);

"and submit the transaction to a database system separate from the first system for execution" as (col. 1, lines 45-67; col. 2, lines 1-10; col.3, lines 45-55).

Ganesh does not explicitly teach the claimed limitations "re-ordering to cause modification operations on the first base relation of the join view to be scheduled for execution, and to cause modification operations on the second base relation to be queued for execution after completion of the modification operations on the first base relation; wherein the controller is adapted to group the modification operations on the set of one or more tuples of the first base relation into a transaction".

Ganesh43 teaches storing a transaction 804 (a second modification operation) and a transaction 806 (a third modification operation) in a table as a queue (fig. 8).

Ganesh43 teaches Since transaction TXA has a dependent SCN of "0", this transaction is not dependent upon any other transactions, and can be ordered before,after or parallel to any other transaction, subject to the ordering/dependency constraints of these other transactions. Transactions TXB, TXC, and TXE all have dep_SCN values of 5; therefore, these transactions must be scheduled to begin after all other transactions having SCN values of 5 or less have completed and committed (col. 16, lines 66-67; col. 17, lines 1-30). Ganesh43 teaches transaction T5 commits at time 25 having executed the following statements: UPDATE Emp_Table SET Emp_value=Emp_value+1 WHERE emp_name=`Smith`; DELETE FROM Emp_Table WHERE emp_name=`Miller'. Statements are represented as modification operations that are grouped in the transaction T5 (col. 5, lines, 1-20, fig. 2). Transactions 1-5 are scheduled (col. 7, lines 22-40).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ganesh43's above teaching to Ganesh's system in order to prevent conflict when multiple transactions trying to modify database records at the same time, minimize network traffic, achieve maximum performance (Ganesh49, col. 9, line 40-41) and further to allow users to track transactions.

As to claim 20, Ganesh teaches the claimed limitation "wherein the controller is adapted to identify the modification operations on the second base relation as conflicting with modification operations on the first base relation in response to

determining that the modification operations on the second base relation are modifying a different base relation of the join view than the modification operations on the first base relation" as (col. 10, lines 45-67).

6. Claims 9-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Anaya et al (or hereinafter "Anaya") (US 5940828).

As to claim 9, Ganesh does not explicitly teach the claimed limitations "storing pending transactions in plural queues corresponding to respective plural session of the database system; and selecting one of the pending transactions from the queues to schedule for execution in the database system based on whether the one pending transaction conflicts with one or more executing transactions in the database system".

Anaya teaches storing modification transactions in plural queues (figs. 2-10).

Ganesh43 teaches scheduling transactions (col. 17, lines 1-30).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Anaya's teaching of storing modification transactions in plural queues and Ganesh43's teaching of scheduling transactions to avoid conflicts between transactions to Ganesh's system in order to minimize transactions.

As to claim 10, Ganesh teaches the claimed limitation "determining that the one pending transaction conflicts with the one or more executing transactions in response to

determining that the one pending transaction modifies a different one of the base relations of the join view than a base relation of the join view modified by an executing transaction" as (col. 10, lines 45-67).

7. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Anaya et al (or hereinafter "Anaya") (US 5940828) and Roffe et al (or hereinafter "Roffe") (US 5442785).

As to claim 11, Ganesh does not explicitly teach the claimed limitation "applying a technique to prevent starvation of one of the pending modification operations in response to determining that the one pending modification operation has been in one of the queues for longer than predetermined time period".

Roffe teaches FIG. 16 is a flow chart for the Timeout Function that checks the Message Response Wait Queue for suspended application programs which are waiting for response messages. The Timeout Function is used to detect processes which have been waiting for an extended period of time for one or more responses. When a program has been suspended for longer than a predetermined period of time, theTimeout Function will resume execution of the suspended program (fig. 16).

It would have been obvious to a person of an ordinary skill in the art at the time invention was made to apply Roffe's teaching of the Timeout Function that checks the Message Response Wait Queue for suspended application programs which are

waiting for response messages. The Timeout Function is used to detect processes, which have been waiting for an extended period of time for one or more responses. When a program has been suspended for longer than a predetermined period of time, theTimeout Function will resume execution of the suspended program to Ganesh's system in order to avoid deadlock situations and data corruption.

8. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and Anaya et al (or hereinafter "Anaya") (US 5940828).and further in view of Goedken (US 2002/0133494).

As to claim 11, Ganesh does not explicitly teach the claimed limitation "applying a technique to prevent starvation of one of the pending modification operations in response to determining that the one pending modification operation has been in one of the queues for longer than predetermined time period".

Goedken teaches the queue manager 134 determines if the current message has been pending for longer than a predetermined period of time. Preferably, the queue manager 134 make this determination by cooperating with the message mapper 126 to subtract the value of the "Asked" timestamp field in the message map database 118 from the current date and by comparing the result to a predetermined time period (e.g., 10 days). The predetermined time period may be fixed for all messages or it may be defined by the information request message 18 (paragraph [0191]).

It would have been obvious to a person of an ordinary skill in the art at the time invention was made to apply Goedken's teaching of the queue manager 134 determines if the current message has been pending for longer than a predetermined period of time to Ganesh's system in order to avoid deadlock situations and data corruption.

9. Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Cochrane et al (or hereinafter "Cochrane") (US 6581205).

As to claim 18, Ganesh teaches the claimed limitations:

"in response to a particular one of modification operations to modify one of the base relations, placing an exclusive lock on the one base relation" as (col. 9, lines 20-30).

Ganesh does not explicitly teach the claimed limitation" placing a predefined lock on the join view, the predefined lock conflicting with each of a shared lock and an exclusive lock placed on the join view but the predefined lock not conflicting with another predefined lock placed on the join view".

Cochrane teaches placing a U-lock on the record in the materialized view. The U-lock conflicting with shared lock (col. 9, lines 50-60).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Gochrane's teaching of placing a U-lock on the record in the materialized view. The U-lock conflicting with shared lock to Ganesh's system in

order to avoiding deadlocks with other transactions that modifying at least one base table of the materialized view and to improve concurrency with other transactions.

10. Claim 21 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Ngai et al (or hereinafter "Ngai") (US 6574717).

As to claim 21, Ganesh teaches the claimed limitations:

"identify modification operations on the first base relation that modify distinct portions of the first base relation" as (col. 4, lines 50-60);

Ganesh does not explicitly teach the claimed limitations "a first system and wherein the controller is adapted to open plural sessions with a database system separate from the first system, submit the identified the modification operations that modify distinct portions of the first base relation through different sessions for concurrent execution in the database system".

Ngai teaches in step 232 the number of sessions is determined. In one embodiment involving a licensed database system, the number of sessions is the number of users who may use a database at one time according to the license for the database system. In another embodiment, the number of sessions is a system parameter determined during configuration to limit the number of concurrent users for performance reasons. The number of sessions is determined because the number of transactions executed concurrently by an instance, and consequently the

amount of undo storage space used, is expected to depend on the maximum number of concurrent users (col. 14, lines 15-30).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ngai's teaching of the number of sessions is determined. In one embodiment involving a licensed database system, the number of sessions is the number of users who may use a database at one time according to the license for the database system. In another embodiment, the number of sessions is a system parameter determined during configuration to limit the number of concurrent users for performance reasons. The number of sessions is determined because the number of transactions executed concurrently by an instance, and consequently the amount of undo storage space used, is expected to depend on the maximum number of concurrent users to Ganesh's system in order to allow resources to be recycled and allocated for new uses by other entities in a computer system, but also guarantee the resources are retained in a given state for consistent use by other entities, even after the entity terminates that first had the resource allocated and prevent network traffic when two transaction assigned in the same session.

11. Claim 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Garth et al (or hereinafter "Garth") (US 6678701).

As to claims 23, Ganesh does not explicitly teach the claimed limitation "wherein the controller comprise a load utility to submit the modification operations to a database system".

Garth teaches a load utility (col. 1, lines 60-65).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Garth's teaching of a load utility to Ganesh's system in order to execute all operations in a database without conflicting.

12. Claims 24-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Garth et al (or hereinafter "Garth") (US 6678701), and Desai et al (or hereinafter "Desai") (US 6567816).

As to claim 24, Ganesh does not explicitly teach the claimed limitation "a continuous load utility".

Desai teaches load utility have to extract the data from the columns in the record that correspond to the index key and then add such data to the index columns (col. 5, lines 45-50).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Desai's teaching of load utility have to extract the data from the columns in the record that correspond to the index key and then add such

data to the index columns to Ganesh's system in order to extract the data from a database.

As to claim 25, Ganesh does not explicitly teach the claimed limitation "the load utility comprise a first load utility, and the controller comprises a second load utility to concurrently submit other modification operations to the database system".

Garth teaches load utilities (col. 1, lines 60-65).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Garth's teaching of a load utility to Ganesh's system in order to load operations for scheduling executing all operations in a database without conflicting.

13. Claim 26 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ganesh et al (or hereinafter "Ganesh") (US 6353828) in view of Ganesh et al (or hereinafter "Ganesh43") (US 6714943) and further in view of Garth et al (or hereinafter "Garth") (US 6678701), Desai et al (or hereinafter "Desai") (US 6567816) and Papierniak et al (or hereinafter "Papierniak") (US 6151601).

As to claim 26, Ganesh does not explicitly teach the claimed limitation "plural platforms on which corresponding first and second load utilities are executable".

Papierniak teaches platforms for corresponding to load utilities (abstract, col. 9, lines 15-20).

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to apply Ganesh's teaching of platforms for corresponding to load utilities to Ganesh's system in order to improve executing load utilities quickly without traffic.

Conclusion

14. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact Information

15. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Cam Y T. Truong whose telephone number is (571) 272-4042. The examiner can normally be reached on Monday to Firday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Cam Y Truong
Primary Examiner
Art Unit 2162